Delivering the Full Potential of PCIe Storage

Amber Huffman
Sr Principal Engineer
Storage Technologies Group
Intel Corporation

August 25, 2013
Agenda

• Architecting from the ground up for NVM

• The Standard Software Interface: NVM Express

• Future Innovation

• Summary
PCIe* for Datacenter/Enterprise SSDs

- PCI Express* is a great interface for SSDs
  - Stunning performance: 1 GB/s per lane (PCIe Gen3 x1)
  - With PCIe scalability: 8 GB/s per device (PCIe Gen3 x8) or more
  - Lower latency: Platform+Adapter: 10 µsec down to 3 µsec
  - Lower power: No external SAS IOC saves 7-10 W
  - Lower cost: No external SAS IOC saves ~ $15
  - PCIe lanes off the CPU: 40 Gen3 (80 in dual socket)

PCIe SSDs are emerging in Datacenter/Enterprise, co-existing with SAS & SATA depending on application.

*Other names and brands may be claimed as the property of others.
Many candidate next generation NVM technologies. Offer ~ 1000x speed-up over NAND, closer to DRAM speeds.
Fully Exploiting Next Gen NVM Requires Platform Improvements

- With Next Gen NVM, the NVM is no longer the bottleneck
  - Need optimized platform storage interconnect
  - Need optimized software storage access methods
Transformation Required

- Transformation was needed for full benefits of multi-core CPU
  - Application and OS level changes required

- To date, SSDs have used the legacy interfaces of hard drives
  - Based on a single, slow rotating platter.

- SSDs are inherently parallel and next gen NVM approaches DRAM-like latencies

- For full SSD benefits, must architect for NVM from the ground up

NVM Express is the interface architected for NAND today and next generation NVM.
Agenda

• Architecting from the ground up for NVM

• *The Standard Software Interface: NVM Express*

• Future Innovation

• Summary
NVM Express

- NVM Express (NVMe) is the standardized high performance host controller interface for PCIe* SSDs

- NVMe was architected from the ground up for non-volatile memory, scaling from Enterprise to Client
  - The architecture focuses on latency, parallelism/performance, and low power
  - The interface is explicitly designed with next generation NVM in mind

- NVMe was developed by an open industry consortium of 90+ members and is directed by a 13 company Promoter Group

*Other names and brands may be claimed as the property of others.
Technical Basics

- All parameters for 4KB command in single 64B command
- Supports deep queues (64K commands per queue, up to 64K queues)
- Supports MSI-X and interrupt steering
- Streamlined & simple command set optimized for NVM (13 required commands)
- Optional features to address target segment of product in Client or Enterprise
  - Enterprise: End-to-end data protection, reservations, etc
  - Client: Autonomous power state transitions, etc
- Designed to scale for next generation NVM, agnostic to NVM type used
**Queuing Interface**

*Command Submission & Processing*

1. Host writes command to Submission Queue
2. Host writes updated Submission Queue tail pointer to doorbell

**Command Processing**

3. Controller fetches command
4. Controller processes command
**Command Completion**

5. Controller writes completion to Completion Queue
6. Controller generates MSI-X interrupt
7. Host processes completion
8. Host writes updated Completion Queue head pointer to doorbell
**Simple Optimized Command Set**

<table>
<thead>
<tr>
<th>Admin Commands</th>
<th>NVM Admin Commands</th>
<th>NVM I/O Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create I/O Submission Queue</td>
<td>Format NVM (optional)</td>
<td>Read</td>
</tr>
<tr>
<td>Delete I/O Submission Queue</td>
<td><strong>Security Send (optional)</strong></td>
<td>Write</td>
</tr>
<tr>
<td>Create I/O Completion Queue</td>
<td><strong>Security Receive (optional)</strong></td>
<td>Flush</td>
</tr>
<tr>
<td>Delete I/O Completion Queue</td>
<td></td>
<td><strong>Write Uncorrectable (optional)</strong></td>
</tr>
<tr>
<td>Get Log Page</td>
<td></td>
<td><strong>Compare (optional)</strong></td>
</tr>
<tr>
<td>Identify</td>
<td></td>
<td><strong>Dataset Management (optional)</strong></td>
</tr>
<tr>
<td>Abort</td>
<td></td>
<td><strong>Write Zeros (optional)</strong></td>
</tr>
<tr>
<td>Set Features</td>
<td></td>
<td><strong>Reservation Register (optional)</strong></td>
</tr>
<tr>
<td>Get Features</td>
<td></td>
<td><strong>Reservation Report (optional)</strong></td>
</tr>
<tr>
<td>Asynchronous Event Request</td>
<td></td>
<td><strong>Reservation Acquire (optional)</strong></td>
</tr>
<tr>
<td><strong>Firmware Activate (optional)</strong></td>
<td></td>
<td><strong>Reservation Release (optional)</strong></td>
</tr>
<tr>
<td><strong>Firmware Image Download (opt)</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Only 10 Admin and 3 I/O commands required.
Proof Point: NVMe Latency

- NVMe reduces latency overhead by more than 50%
  - SCSI/SAS: 6.0 µs 19,500 cycles
  - NVMe: 2.8 µs 9,100 cycles

- Increased focus on storage stack / OS needed to reduce latency even further
NVM Express Deployment Beginning

- NVM Express 1.0 specification published in March 2011
  - Additional Enterprise and Client capabilities included in NVMe 1.1 (Oct 2012)

- First plugfest held May 2013 with 11 companies participating
  - Interoperability program run by University of New Hampshire Interoperability Lab, a leader in PCIe*, SAS, and SATA compliance programs

FOR IMMEDIATE RELEASE

NVM Express Workgroup Holds First Plugfest

Milestone in Process to Deliver Standards-based Interoperability for PCI Express Solid-State Drives

WAKEFIELD, Mass., May 29, 2013 — The NVM Express Workgroup, developer of the NVM Express specification for accessing solid-state drives (SSDs) on a PCI Express (PCIe) bus, held its first Plugfest at the University of New Hampshire InterOperability Lab in Durham, N.H., May 13-16, 2013. This event provided an opportunity for participants to measure their products’ compliance with the NVM Express (NVMe) specification and to test interoperability with other NVMe products.

The NVMe specification defines an optimized register interface, command set and feature set for PCIe-based Solid-State Drives (SSDs). NVMe refers to non-volatile memory, as used in SSDs. The goal of NVMe is to unlock the potential of PCIe SSDs now and in the future, and to standardize the PCIe SSD interface. Participating in the Plugfest were Agilent Technologies, Dell Inc., Fastor Systems, Inc., HGST, a Western Digital company, Integrated Device Technology, Inc., Intel Corporation, Samsung Electronics Co., Ltd., SanDisk Corporation, sTec, Inc., Teledyne LeCroy, and Western Digital Corporation.

NVMe products targeting Datacenter shipping later this year.

Samsung has announced the XS1715, a 2.5-inch Non-Volatile Memory Express (NVMe) PCIe SSD. According to Samsung, the 1.6TB SFF-8639 NVMe SSD provides a sequential read speed at 3,000MB/s, six times faster than the company’s current high-end enterprise SSD. The XS1715’s random read performance is specified at up to 740,000 IOPS, more than 10 times as fast as existing SSD options.
Agenda

• Architecting from the ground up for NVM
• The Standard Software Interface: NVM Express
• *Future Innovation*
• Summary
Storage Programming Model
Innovation Needed

• We’re starting to outgrow the block storage model
  – Memory like attributes on Next Gen NVM
    – Next Gen NVM small granularity accesses
    – Next Gen NVM near memory speeds
  – Even fast NAND based SSDs are held back today

• Possibilities:
  – Lower Latency Stack (partially addressed by NVMe)
  – Kernel Bypass?
  – Persistent Memory?

“Memory like” attributes possible with Next Gen NVM. New programming models are needed to take full advantage.
SNIA NVM Programming TWG

- SNIA NVM Programming Technical Workgroup founded June 2012
  - Founders: Dell, EMC, Fujitsu, HP, IBM, Intel, NetApp, Oracle, QLogic, Symantec
  - Many active members: Intel, HP, Microsoft, Fusion-IO, etc.

- Charter: Develop specifications for new software “programming models” as NVM becomes a standard feature of platforms

- Scope of TWG work includes:
  - In-kernel NVM programming models
  - Kernel-to-application programming models
  - New NVM “memory usage models”

- OS Specific APIs
  - SNIA defines the programming model specification
  - Each OSV codes the programming models to specific to OS
    - E.g.,: Open Source project underway to provide the Linux* implementation of effort

SNIA specifications + OS implementations defines solution.
Programming Model Stack Diagram

1. NVM as Storage
   - Existing Applications
     - Management Applications (GUI, CLI, CIM)
   - NVM Management API

2. Application Optimization
   - NVM-Optimized Applications
   - NVM User-Space API

3. NVM as Memory
   - "Persistent Memory"

**NVM Driver Stack**
**NVM Kernel API**
**NVM-Optimized Kernel Modules**

**User**
**Kernel**
**Hardware**

SNIA NVM Programming TWG
Linux* Open Source Project, Microsoft*, other OSVs
Existing/Unchanged Infrastructure

*Other names and brands may be claimed as the property of others.
Agenda

• Architecting from the ground up for NVM
• The Standard Software Interface: NVM Express
• Future Innovation
• Summary
Summary: Transformation Required

Recall:

- Transformation was needed for full benefits of multi-core CPU
  - Application and OS level changes required

- To date, SSDs have used the legacy interfaces of hard drives
  - Based on a single, slow rotating platter..

- SSDs are inherently parallel and next gen NVM approaches DRAM-like latencies

To realize the full potential of PCIe* SSDs, architect from the ground up for NVM.

*Other names and brands may be claimed as the property of others.
PCIe* Storage Usage Models

**Server Caching**
- Used for temporary data
- Non-redundant
- Used to reduce memory footprint

**Server Storage**
- Typically for persistent data
- Redundant (i.e., RAID’ed)
- Commonly used as Tier-0 storage

**Client Storage**
- Used for Boot/OS drive and/or HDD cache
- Non-redundant
- Power optimized

**External Storage**
- Used for just metadata or all data
- Multi-ported device
- Redundancy based on usage

*Other names and brands may be claimed as the property of others.
Form Factor & Connector Landscape

Ultrabook™ Mobile All-in-one Desktop WS Server

- CEM add-in card supports high speed SSDs with 4, 8, 16 lanes of PCIe*
- M.2 is designed for Ultrabook™ with PCIe x2 or SATA, or PCIe x4
- SFF-8639 designed for Enterprise use – supports 2.5” PCIe x4, SAS, SATA
- 2.5” SATA Express connector designed for client with 2 lanes PCIe or SATA

*Other names and brands may be claimed as the property of others.
Proof Point: NVMe in a SAN

- Demo combines NVMe with existing ingredients to deliver > 3.1M random 4K IOPs
- The performance of direct attached (DAS) NVMe SSDs married to an FCoE SAN
- Next generation SAN is possible today by use of highly efficient interfaces

SAN with NVMe:
3.1 Million random 4K IOPs on 120Gbps FCoE.

Storage target configuration: Intel® S2600IP4 Server Board, Intel® Xeon® Processor E5-2690 2.9GHz, 8-16GB DDR3 1033 DIMMs, RH EL-6.2 – 3.3.0-RC1 kernel, TCM storage target, 4 Intel® Ethernet Server Adapter X520 (10 Gbps CNA).
Initiator configuration: 12 initiators: Intel® Xeon® Processor 5650 2.67GHz, RH EL-6.2 – 3.3.0-RC1 kernel.
Test configuration: (per initiator) Linux fio V21.0.7, 4K Random Read, QD=8, Workers=16, 8 FCoE LUNs.